Algoritmo de Fleury

por

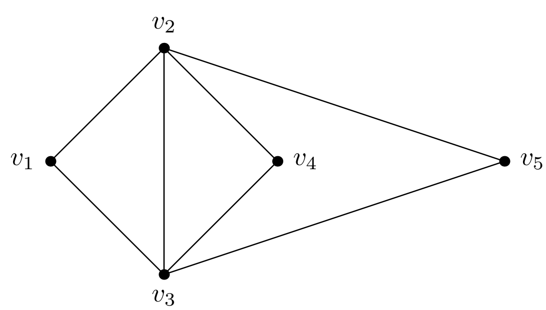
Ramón Espinosa Armenta

El siguiente algoritmo, debido a Fleury (1921), permite construir un circuito Euleriano en un multigrafo Euleriano.

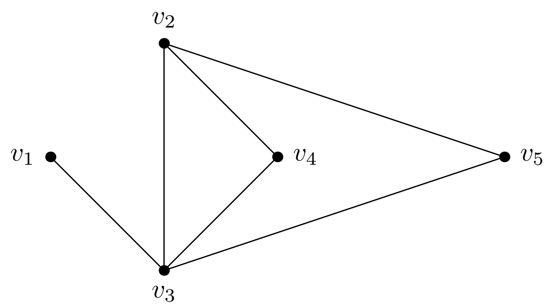
Algoritmo Fleury (G)
Entrada. Un multigrafo Euleriano G.
Salida. Un paseo P en G.

1. Elegir un vértice arbitrario $v_1 \in V$. Definir $P_0 = (v_1)$, $G_0 = G$.
2. Si el paseo $P_k = (v_1, e_1, v_2, \ldots, e_{k-1}, v_k)$ ha sido construido,
   elegir $e_k \in E(G_{k-1})$ tal que
   (i) e_k incide con v_k.
   (ii) a menos que no haya otra elección e_k no es un puente de G_{k-1}.
   Si no existe tal k alto.
3. Definir $P_k := (P_{k-1}, e_k, v_{k+1})$, donde v_{k+1} es el otro extremo de e_k.
   Definir $G_k := G_{k-1} - e_k$.
4. Devolver $P := P_k$.

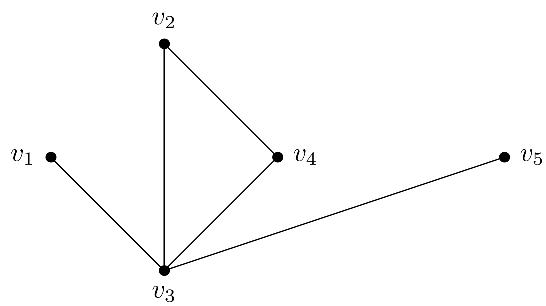
Ejemplo 1. Utilizar el algoritmo de Fleury para construir un paseo en el grafo:



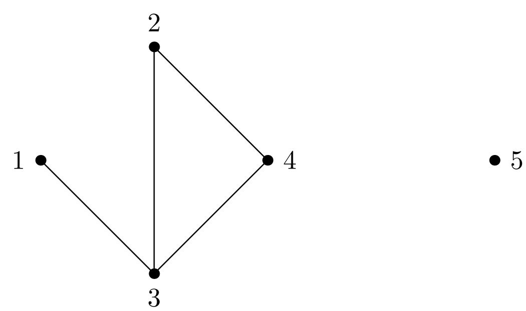
Solución. Comencemos con el vértice v_1. La arista v_1v_2 no es un puente, por lo que podemos elegir $e_1 = v_1v_2$. La siguiente figura muestra el grafo $G_1 = G - e_1$.



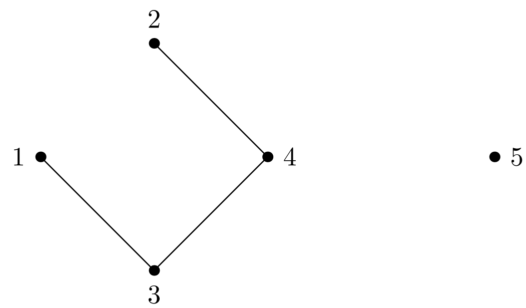
La arista v_2v_5 no es un puente, por lo que podemos elegir $e_2 = v_2v_5$. La siguiente figura muestra el grafo $G_2 = G_1 - e_2$.



La arista v_5v_3 es un puente, pero no tenemos otra elección posible, por lo que definimos $e_3 = v_5v_3$. La siguiente figura muestra el grafo $G_3 = G_2 - e_3$.



La arista v_3v_2 no es un puente, por lo que podemos elegir $e_4 = v_3v_2$. La siguiente figura muestra el grafo $G_4 = G_3 - e_4$.



A partir de aquí es claro que debemos elegir $e_5 = v_2v_5$, $e_6 = v_5v_3$, $e_7 = v_3v_1$. De esta manera el paseo está descrito por la sucesión de vértices:

$$P = (v_1, v_2, v_5, v_3, v_2, v_4, v_3, v_1).$$

$\triangle$

Teorema 1. *Si G es un multigrafo Euleriano, y P es un paseo construido por el algoritmo de Fleury, entonces P es un circuito Euleriano.*

Demostración. Sea $P = (v_1, e_1, v_2, \ldots, e_r, v_r)$ un paseo construido por el algoritmo de Fleury.

Claramente el grado de v_r en $G_r = G - \{e_1, \ldots, e_r\}$ es cero, porque si no fuera así el algoritmo no se habría detenido.

Cada vértice distinto de v_1 y v_r es incidente con un número par de aristas de P_r, por lo que también tiene grado par en G_r. Si $v_r \neq v_1$ entonces cada vez que aparece v_1 en P_r después de la primera vez debe ser como vértice interior, de modo que $d_{G_r}(v_r)$ debe ser impar. Entonces v_r es el único vértice de grado impar en G_r lo cual no es posible. La contradicción surgió de suponer que $v_r \neq v_1$, por lo que $v_r = v_1$ y P_r es un paseo cerrado.

Supongamos ahora que P_r no es un circuito Euleriano y sea

$$S = \{v \in V(G_r) \mid d_{G_r}(v) > 0\}.$$

Por lo tanto S es no vacío y $v_0 = v_r \in T = V - S$. Sea k el mayor entero tal que $v_k \in S$ y $v_{k+1} \in T$. Como P_r termina en T, e_{k+1} es la única arista en $G_k = G - \{e_1, \ldots, e_k\}$ con un extremo en S y un extremo en T. Por lo tanto e_{k+1} es un puente en G_k. Sea e cualquier otra arista en $G - k$ incidente con v_k. Por lo tanto e debe ser un puente de G_k y por lo tanto también de $G_k[S]$. Pero como $G_k[S] = G_r[S]$, cada vértice en $G_k[S]$ es de grado par, lo cual no es posible, pues $G_k[S]$ no tiene puentes. Por lo tanto P_r es un circuito Euleriano. $\square$

El problema chino del cartero

Cada día laboral, un cartero sale de la oficina postal y recorre todas las calles de su zona para entregar el correo y luego regresa a la oficina postal. El cartero desea encontrar la ruta que le permita caminar lo menos posible.

El problema anterior es conocido como el **problema chino del cartero**, pues fue propuesto en 1962 por el matemático chino *Kuan*[1]. En términos matemáticos el problema es el siguiente:

Sea G un multigrafo conexo y sea $c : E(G) \to \mathbb{R}$, con $c(e) \geq 0$ para toda $e \in E(G)$. Hallar un camino cerrado:

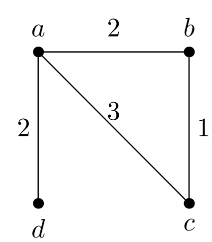
$$W = v_1 e_1 v_2 \dots e_r v_{r+1}$$

que pase por todas las aristas de G y tal que

$$c(W) = \sum_{i=1}^{r} c(e_r)$$

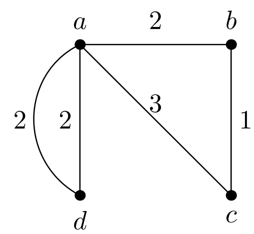
sea mínima.

Si G es Euleriano entonces un circuito Euleriano es una solución óptima para el problema, en otro caso algunas aristas se recorren más de una vez, por ejemplo, en el grafo:



un camino óptimo es a, b, c, a, d, a, aquí la arista $\{a, d\}$ se recorre dos veces. Si duplicamos esta arista obtenemos el grafo Euleriano:

[1] M. K. Kuan (1962). Graphic programming using odd or even points. *Chinese Math.*, **1**, 273-277.



Esta observación nos conduce a plantear el siguiente problema.

Dado un grafo conexo G y una función no negativa $c : E(G) \to \mathbb{R}$, hallar, duplicando aristas, un multigrafo Euleriano G^* tal que

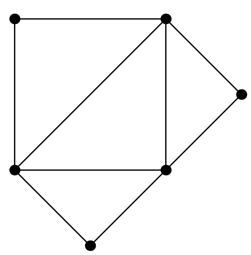
$$\sum_{e \in E(G^*) - E(G)} c(e)$$

sea lo más pequeña posible.

Observemos que un circuito Euleriano en G^* es una solución óptima para el problema chino del cartero. En 1973 *Edmonds y Johnson*[2] diseñaron un algoritmo polinomial para resolver el problema chino del cartero, sin embargo su solución es demasiado complicada para ser presentada aquí.

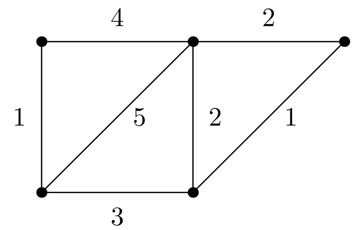
Ejercicios

1. Utiliza el algoritmo de Fleury para hallar un circuito Euleriano en el grafo:



[2] J. Edmonds y E. L. Johnson. (1973). Matching, Euler tours and the Chinese postman. *Math. Programming*, **5**, 88-124.

2. Resuelve el problema chino del cartero para el grafo:



3. Determina la complejidad computacional del algoritmo de Fleury.